Allied Telesis

Best Practice | On-Site Debugging

Introduction

This How To Note describes techniques for on-site debugging. It is in no way intended to be a comprehensive guide to debugging Allied Telesis equipment. Instead, it is aimed at passing on some techniques that have been found to be useful. Particularly, this document describes techniques which:

- make it less likely that you will miss important pieces of information
- make it simpler for someone else to analyse the debug information that you capture
- enable you to be more accurate in after-the-event discussions of what you saw.

Which products does it apply to?

This document applies to the following Allied Telesis routers and managed layer 3 switches:

- AR300, AR400, and AR700 series routers
- AT-8600, AT-8700XL, AT-8800, Rapier, and Rapier i series switches
- AT-9800 series switches
- AT-8900, AT-9900, and AT-9900s series switches
- x900 series switches
- SwitchBlade series switches

Related How To Notes

You also may find the following How To Notes useful, because they specifically discuss how to debug connections:

- *How To Troubleshoot ISDN Connections*
- *How To Troubleshoot VPNs*

Note that many other How To Notes include troubleshooting information as well as configuration examples.

How To Notes are available from www.alliedtelesis.com/resources/literature/howto.aspx.

Starting off

When going on-site to investigate a network problem, your guiding aim is to understand the root cause of the problem, which might be:

- a cable plugged into the wrong socket
- a misconfiguration of some piece of equipment
- an interoperability problem between two pieces of equipment
- a hardware fault in some piece of equipment
- a bug in the software of some piece of equipment.

Of course, it might not always be possible to find the root cause, but looking for it should be the central target of your debugging session.

However, it is also important to make sure that you are also capturing enough information (and the right sort of information) to enable someone else to effectively analyse what you have captured. This could be so that they can confirm your theory about the root cause of the problem, **or**, in the case that you were not able to nail down the root cause, so that they can develop their own theories about what the cause might be.

Concrete piece of advice #1: Capture everything

Right from the first moment you attach your terminal emulator to the console port of the first piece of equipment, ensure that the logging facility of the terminal emulator is turned on. This means that you will capture a lot of (seemingly) irrelevant stuff, but it also means that you will not fail to capture the important stuff.

Concrete piece of advice #2: Capture the initial state

It is important to approach the problem with an open mind, and start with a general survey of the equipment you are looking at, to give an initial picture of how things were at the start of debugging session. This might be the "network operating OK" state, or the "things going wrong" state, depending on what state the network was in when you arrived.

In this initial survey, you should capture a **show debug** from every relevant unit, and capture all the counters and module states that are relevant to the issue.

For example, if the network consists of switches, and the problem is related to PIM multicast routing, then you will want to capture the following commands:

```
show switch count
show switch port=all
show switch port=all count
show ip route
show ip route multi
show ip igmp
```

```
show ip count
show pim route
show pim count
show pim neighbour
```

and other commands relevant to the hardware forwarding of multicast in the particular models of switch you are working with.

Once this initial survey is complete, you can start following up your theories about what is going on.

Following up theories

Starting from the symptoms of problem, you can work back through the steps in the networking process to think of various points where something could have gone wrong, and follow up on each of those in turn.

For example, if the symptom is that a certain client never receives multicast streams, then the problem could be:

- the client is failing to send the IGMP report; or
- the client is sending the report, but it is being dropped by a switch between the client and the device running PIM; or
- the IGMP report is reaching the PIM router, but a multicast route is not being created; or….

So, the approach is to follow each of these possibilities—capturing counters, **enable debug** outputs, ethereal traces, etc to confirm or deny each one. When a particular possibility looks fruitful, then dig deeper into it to try and narrow down more accurately what is not happening correctly. For example, if you see that the IGMP report is not being seen by a particular switch in the chain, you might be able to find that an error counter in the **show switch port count** output increments every time you try sending the IGMP report, indicating that the reports are being corrupted.

> **Concrete piece of advice #3: Type your thoughts and actions.**

It is important to always keep in mind that it is quite likely that your debug capture will be analysed by someone else later on (or, in fact, you might need to go back through it later yourself). Therefore, it is necessary to record your thinking and actions as you were going along, to give context to all the debugging that was captured.

This achieves two main aims:

- it makes it clear **why** you were looking at particular counters etc
- it makes it clear how the captures relate to external events. So often it is **vital** to know whether a particular state change or counter change seen in the debug capture happened before or after a particular external event. For example, 'did that switch port on blade 6 go into STP blocking state before or after port 24 on blade 5 was unplugged..'.

All the way through the capture, type (just at the CLI prompt) what you are thinking, and what you are doing. Just write it as stream-of-consciousness, things like:

"Hmmm.. that counter value looks strange, lets look at this a bit more"

"I have just pulled out blade 5, lets look the internal ports…"

"Right, the ping to the PC on port 5 just starting failing ".

Of course, your consciousness might not stream in English. If so, then write the comments in your language; the important ones can always be translated later. The important thing is that the writing of the comments does not get in the way of the job at hand.

Of course, writing comments at the CLI prompt will mean that you get error messages from the unit like:

```
Manager > Now I am going to send the IGMP report from the client,
  and see if any PIM counters or states change

Error (3035256): Unknown command "Now".
```

But that is quite OK, the person analysing the capture will be able to mentally filter out these error messages.

When you see something really significant, you might want to highlight it, so it is easy to find later. A string of !!!!!!!! or ####### are good ways to make sure something stands out when you are browsing through the file.

What counters are changing?

Often, the investigation of a theory requires looking for what is changing—for example, are the error counters on a particular port increasing? Is Port A now sending more multicast packets than Port B is receiving? Are entries in the route ARP table being refreshed? This brings us to concrete piece of advice #4.

Concrete piece of advice #4: Perform particular show commands a few times in succession.

You might see certain values changing that give a vital clue to what is happening. Or, if you don't notice them, the person analysing the capture after you might.

Rate of change can be important

More subtly, the **rate** at which certain things change can often be significant—for example, how long did it take that unit to make the transition from VRRP master to slave? or how long did that route take to age out? or other such questions. Unfortunately, the outputs of **show** commands are not time-stamped, so sometimes it is necessary to do the "time stamping" yourself:

Concrete piece of advice #5: If you are investigating matters that are time-related, then enter a "show time" command before each of your other commands.

Providing a convincing case

If you believe that you have successfully hit on the right theory, and have isolated what appears to be a good candidate for the root cause of what is going wrong, gather as much evidence as you can to support the theory.

> **Concrete piece of advice #6: If you find a sequence of actions that make the problem happen, capture the full sequence (and the evidence that the problem has happened) multiple (more than 2) times if possible.**

Often certain pre-conditions are pertinent to the circumstances in which the problem will or won't occur—'the problem still occurs even if port 2 is not connected'; 'if the ARP entry has been cleared before the route ages out, the problem does not happen' etc. It is important to capture the output to illustrate these pre-conditions—it might be very clear to you that port 2 is definitely disconnected, but the person analysing the captures after you might not truly believe that unless they see the output of the **show port** command that shows port 2 to be DOWN.

> **Concrete piece of advice #7: Capture as much evidence as possible to support any assertions you make about the circumstances in which you see the problem happen.**

Annotating afterwards

Before you send your capture off to the next layer of support or anyone else, you will want to give them as much guidance as possible as to how to extract the significant pieces of information from your capture. This can be achieved by going through the capture and putting in further comments to spell out more clearly what is happening than you were able to do with the quick comments you typed in while on-site. Or, you might want to copy the really important parts out of the capture and paste them into a separate file, to succinctly illustrate the core of the problem.

> **Concrete piece of advice #8: Some post-processing of the capture file can save a lot of time and effort for anyone who is analysing it after you.**

But, even if you copy the pertinent pieces out of the capture, still send the rest of the capture file anyway, there could be significant gems of information in there that you weren't aware of.

> **Concrete piece of advice #9: If possible, always send everything that you captured—too much information is always better than too little.**

USA Headquarters | 19800 North Creek Parkway | Suite 200 | Bothell | WA 98011 | USA | T: +1 800 424 4284 | F: +1 425 481 3895
European Headquarters | Via Motta 24 | 6830 Chiasso | Switzerland | T: +41 91 69769.00 | F: +41 91 69769.11
Asia-Pacific Headquarters | 11 Tai Seng Link | Singapore | 534182 | T: +65 6383 3832 | F: +65 6383 3830
www.alliedtelesis.com

C613-16028-00 REV B